

# Asymmetrically Weighted CCA And Hierarchical Kernel Sentence Embedding For Multimodal Retrieval

Youssef Mroueh  
Etienne Marcheret  
Vaibhava Goel

MROUEH@US.IBM.COM  
ETIENNEM@US.IBM.COM  
VGOEL@US.IBM.COM

Multimodal Algorithms and Engines Group, IBM T.J Watson Research Center, USA.

## Abstract

Joint modeling of language and vision has been drawing increasing interest. A multimodal data representation allowing for bidirectional retrieval of images by sentences and vice versa is a key aspect. In this paper we present three contributions in canonical correlation analysis (CCA) based multimodal retrieval. Firstly, we show that an asymmetric weighting of the canonical weights, while achieving a cross-view mapping from the search to the query space, it improves the retrieval performance. Secondly, we devise a computationally efficient model selection - crucial to generalization and stability - in the framework of the *Bjork Golub* algorithm for regularized CCA via spectral filtering. Finally, we introduce a Hierarchical Kernel Sentence Embedding (HKSE) that approximates Kernel CCA for a special similarity kernel between words distributions. State of the art results are obtained on MSCOCO and Flickr benchmarks when these three techniques are used in conjunction.

## 1. Introduction: Multimodal Retrieval

Modeling jointly language and vision has attracted a lot of attention recently. Generative models such as deep recurrent networks for the language modeling, in conjunction with deep convolutional neural networks on the image side have shown remarkable success in the image captioning task (Karpathy & Li, 2015; Mao et al., 2014; Vinyals et al., 2015; Socher et al., 2011). Image and Text retrieval has been the focus of many recent works. (Fang et al., 2015; Klein et al., 2015; Lebrete et al., 2015; Kiros et al., 2015; 2014; Karpathy et al., 2014; Gong et al., 2014;

Socher et al., 2014; Kulkarni et al., 2011; Farhadi et al., 2010). The main contributions of this paper are :

**1) Mapping search items to the query space, AW-CCA.** In multimodal retrieval it is more common to embed the search space and the query space into a shared space (CCA in (Klein et al., 2015)) or to map the query item to the search space (cross-view mapping in (Socher et al., 2013)). In this paper we empirically show that mapping the search items to the query space outperforms those standard methods. We further show that this cross-view mapping from the search to the query space can be implemented by a simple asymmetric weighting of the canonical correlation weights (AW-CCA), where the canonical weights of the search space are weighted by the canonical correlations.

**2.a) Regularization and Spectral Filtering for CCA.** Regularization is a key factor for the numerical stability as well as the generalization properties of a learning algorithm. We revisit Regularized CCA (Vinod, 1976) within the framework of spectral filtering and the Bjork Golub Algorithm (Golub et al., 1973) (Algorithm 1 in Appendix C). We present our regularized CCA within two spectral filtering regularization families: Tikhonov regularization (Vinod, 1976) and truncated SVD (T-SVD) regularization (Hansen, 1986). T-SVD regularization is new in the CCA context.

**2.b) Fast T-SVD guided Tikhonov cross-validation for CCA.** We show that the truncated SVD CCA regularization path<sup>1</sup> can be computed more efficiently than the Tikhonov regularization path, at the price of a small loss in accuracy. In light of the spectral filtering interpretation we propose a hybrid algorithm that takes advantage of the fast computation of the regularization path of T-SVD CCA, in choosing a regularization parameter for the Tikhonov counterpart (Algorithm 4 in Appendix C), enabling a computationally lightweight exhaustive model selection, thanks to this hy-

<sup>1</sup>Regularization path refers to the model obtained for different regularization parameters as done in cross-validation (Friedman et al, 2010).

brid strategy.

**3) Hierarchical Kernel Sentence Embedding and State of the art results.** We propose the Hierarchical Kernel Sentence Embedding (HKSE) as a means for aggregation of words embeddings (word2vec), that explains and outperforms the mean word2vec baseline. Using those features and mapping the search items to the query space via the asymmetric weighting of the Regularized CCA (T-SVD guided Tikhonov) and the cosine similarity, we achieve state of the art bidirectional retrieval results on the MSCOCO (Lin et al., 2014) and Flickr benchmarks (Hodosh et al., 2013; Young et al, 2014), with off the shelf features for image and text descriptions.

**Notation.**  $\mathcal{Q}$  and  $\mathcal{S}$  are query and search spaces. Given a multimodal training set  $S = \{(x_i, y_i) | x_i \in \mathcal{X} \subset \mathbb{R}^{m_x}, y_i \in \mathcal{Y} \subset \mathbb{R}^{m_y}, i = 1 \dots n\}$ , ( $n > \max(m_x, m_y)$ ), let  $X \in \mathbb{R}^{n \times m_x}$ , and  $Y \in \mathbb{R}^{n \times m_y}$  be the two data matrices corresponding to each modality. Define  $\mu_X, \mu_Y$  to be the means of  $X$  and  $Y$  respectively. Let  $C_{XX} = (X - \mu_X)^\top (X - \mu_X) \in \mathbb{R}^{m_x \times m_x}$ , and  $C_{YY} = (Y - \mu_Y)^\top (Y - \mu_Y) \in \mathbb{R}^{m_y \times m_y}$  be the covariances matrices of  $X$  and  $Y$  respectively. Let  $C_{XY} = (X - \mu_X)^\top (Y - \mu_Y) \in \mathbb{R}^{m_x \times m_y}$  be the correlation matrix. Define  $I_k$  to be the identity matrix in  $k$  dimensions. SVD stands for the *thin* singular value decomposition. A validation set  $S_v$  is given for model selection. Our goal is to index a test set  $S^* = \{(x_i^*, y_i^*) | x_i^* \in S_x^* \subset \mathcal{X}, y_i^* \in S_y^* \subset \mathcal{Y}, i = 1 \dots n^*\}$  for bidirectional search.  $X$  and  $Y$  are assumed to be centered. For  $X$  non singular, let  $X = U\Sigma V^\top$ , and  $C_{XX} = X^\top X$ , we define  $C_{XX}^{-\frac{1}{2}} = V\Sigma^{-\frac{1}{2}}$ .

## 2. Bidirectional Retrieval: Mapping the Search Space to the Query Space

In information retrieval, given a query  $q \in \mathcal{Q}$  and a search item  $s \in \mathcal{S}$  (referred to usually as a document), two probabilistic retrieval approaches are possible (Lafferty et al, 2002): 1) The Search Generation approach, modeling  $\mathbb{P}(s|q)$  (*how likely is a search item given the query*) known as the traditional probabilistic approach 2) The query generation approach, modeling  $\mathbb{P}(q|s)$  (*how likely is a query item given a search item*), known also as the language modeling approach. Lafferty et al showed that while both models are equivalent probabilistically as they are based on a different parametrization of a same joint *relevance* likelihood, they are different statistically as the models are estimated differently. (Lafferty et al, 2002) showed that, whereas the Query Generation approach implicitly models the *relevance* between the query and the search item, the Search generation approach needs an explicit modeling of the relevance by means of positive and negative examples (See Appendix A). The implicit relevance modeling makes

the query generation approach appealing and thus widely used in language modeling. Assuming a gaussian model in the query generation approach,  $q|s \sim \mathcal{N}(Ts, \sigma^2 I)$ , this corresponds to mapping the search space to the query space in a least squares sense ( $\mathcal{S} \rightarrow \mathcal{Q}$ ). Similarly for a gaussian model the search generation approach (without an explicit modeling of the relevance), corresponds to mapping the query space to the search space ( $\mathcal{Q} \rightarrow \mathcal{S}$ ). Another approach for retrieval is the generative model of both queries  $q$  and search items  $s$  from a common hidden variable  $z$ . Under gaussian assumptions for  $q|z, s|z$ , and  $z$ , this corresponds to probabilistic CCA (Bach et al, 2005), which consists of mapping the query and the search space to a shared embedding space  $\mathcal{Z}$  ( $\mathcal{Q} \rightarrow \mathcal{Z} \leftarrow \mathcal{S}$ ). In this paper we follow the query generation approach in the gaussian model i.e *mapping the search space to the query space*, and we show its empirical superiority (See Table 1). We hypothesize that this superiority is due to its implicit modeling of relevance between image and text.

**Bidirectional Retrieval  $\mathcal{S} \rightarrow \mathcal{Q}$ .** We start by defining more formally the bidirectional retrieval tasks. Given pairs of high dimensional points  $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$  where  $x_i$  corresponds to the feature representation of an image given by a deep convolutional neural network, and  $y_i$  a sentence embedding of an associated caption. Our goal is to index this multimodal data in a way that enables bidirectional retrieval: the image annotation task associating a caption to a query image and the image search task associating an image to a query caption. In this Section we use the query generation approach, mapping the search to the query, as discussed earlier, because of its empirical superiority to the search generation approach as found in (Socher et al., 2013), and to the mapping of the search and the query spaces to a shared space (CCA) as in (Klein et al., 2015) (See Table 1 and Appendix D). In order to reduce the vari-

Mapping	Search R@10	Annotation R@10
$\mathcal{Q} \rightarrow \mathcal{S}$	34.61	35.74
$\mathcal{Q} \rightarrow \mathcal{Z} \leftarrow \mathcal{S}$ (CCA)	38.63	47.16
$\mathcal{S} \rightarrow \mathcal{Q}$ (this paper)	<b>44.24</b>	<b>53.12</b>

Table 1: Various mappings performance in bidirectional retrieval on the MSCOCO Benchmark(5K test, VGG+Mean word2vec) (in %).

ance of our estimators we put ourselves in the whitened space of the data in  $X$ , and  $Y$ , i.e we consider the bidirectional mapping between the whitened data spaces (assuming non singularity covariances the whitened data is given by  $XC_{XX}^{-\frac{1}{2}}$ , and  $YC_{YY}^{-\frac{1}{2}}$ , regularization alleviates this issue of non singularity). Hence for the image search task we minimize the following problem:

$$\min_{T \in \mathbb{R}^{m_x \times m_y}} \left\| X C_{XX}^{-\frac{1}{2}} T - Y C_{YY}^{-\frac{1}{2}} \right\|_F^2, \quad (1)$$

The solution of Problem (1) is given simply by  $T = C_{XX}^{-\frac{1}{2}, \top} C_{XY} C_{YY}^{-\frac{1}{2}}$ , this defines the image to caption mapping:  $f_{i \rightarrow c}(x^*) = T^\top C_{XX}^{-\frac{1}{2}, \top} x^*$ . The image search problem, then reduces to transforming the image set through the mapping  $f_{i \rightarrow c}$ , and then finding the nearest neighbor of the query caption represented by the whitened vector  $C_{YY}^{-\frac{1}{2}, \top} y^*$  in the transformed image set. Swapping the roles of  $X$  and  $Y$  we solve equivalently the image annotation problem via a linear least squares. In this case we define equivalently the caption to image mapping:  $f_{c \rightarrow i}(y^*) = T C_{YY}^{-\frac{1}{2}, \top} y^*$ . We can simplify the expressions for image search and image annotation mappings, using the singular value decompositions of  $X$  and  $Y$ . Let  $X = U_x \Sigma_x V_x^\top$  be the SVD of  $X$  and  $Y = U_y \Sigma_y V_y^\top$  be the SVD of  $Y$ . It is easy to show that the whitened data  $X C_{XX}^{-\frac{1}{2}} = X V_x \Sigma_x^{-1} = U_x$ , and  $Y C_{YY}^{-\frac{1}{2}} = Y V_y \Sigma_y^{-1} = U_y$ . For  $x \in \mathcal{X}$  and  $y \in \mathcal{Y}$ , let  $u_x$  and  $u_y$  be the whitened data points:  $u_x = \Sigma_x^{-1} V_x^\top x$ , and  $u_y = \Sigma_y^{-1} V_y^\top y$ . Note that:  $T = C_{XX}^{-\frac{1}{2}, \top} C_{XY} C_{YY}^{-\frac{1}{2}} = \Sigma_x^{-1} V_x^\top X^\top Y V_y \Sigma_y^{-1} = U_x^\top U_y$ .  $T$  corresponds to the correlation in the whitened spaces of  $X$  and  $Y$ . Cross-view mappings are then simply  $f_{i \rightarrow c}(x^*) = T^\top u_x^*$  and  $f_{c \rightarrow i}(y^*) = T u_y^*$ . Hence  $T$  plays a central role in the retrieval problem.

### 2.1. Canonical Correlation Analysis.

We review in this section Canonical Correlation Analysis due to (Hotelling, 1936). For data matrices  $X \in \mathbb{R}^{n \times m_x}$  and  $Y \in \mathbb{R}^{n \times m_y}$ , let  $k = \min(m_x, m_y)$  the canonical correlations  $\sigma_1, \dots, \sigma_k$ , and their corresponding pairs of correlations weights  $\{(u_i, v_i)\}_{i=1 \dots k}$ , given by the columns of  $U \in \mathbb{R}^{m_x \times k}$  and  $V \in \mathbb{R}^{m_y \times k}$ , where  $U$  and  $V$  are the solution of the following maximization problem:

$$\max_{U^\top C_{XX} U = I_k, V^\top C_{YY} V = I_k} \text{Tr}(U^\top C_{XY} V),$$

where  $\sigma_i = u_i^\top C_{XY} v_i$ ,  $i = 1 \dots k$ . Intuitively CCA finds the directions that are maximally correlated and that are orthonormal in the metric defined by each covariance matrix, respectively. The following Lemma due to Bjork and Golub shows that the canonical correlation weights can be computed using the singular value decomposition of the data matrices  $X$  and  $Y$ , and the correlation matrix in the whitened space  $T$  defined in Section 2:

**Lemma 1** ((Golub et al., 1973)). *Let  $X = U_x \Sigma_x V_x^\top$ , and  $Y = U_y \Sigma_y V_y^\top$  be the singular value decomposition of  $X$  and  $Y$  ( $U_x \in \mathbb{R}^{n \times m_x}$ ,  $\Sigma_x \in \mathbb{R}^{m_x \times m_x}$ ,  $V_x \in \mathbb{R}^{m_x \times m_x}$ ). Let  $k = \min(m_x, m_y)$ , and  $T = U_x^\top U_y$ . Let*

$$T = P_x \Sigma P_y^\top \quad (2)$$

*be its SVD,  $P_x \in \mathbb{R}^{m_x \times k}$ ,  $\Sigma \in \mathbb{R}^{k \times k}$ ,  $P_y \in \mathbb{R}^{m_y \times k}$ . The canonical correlations of  $X$  and  $Y$  are the diagonal elements of  $\Sigma$ , with canonical weights of  $X$  given by  $U = V_x \Sigma_x^{-1} P_x$ , and the canonical weights of  $Y$  given by  $V = V_y \Sigma_y^{-1} P_y$ . (Proof in Appendix B).*

Algorithm 1 (Appendix C) summarizes the Bjork Golub procedure to compute CCA. While the original algorithm of Bjork and Golub uses the QR factorization of  $X$  and  $Y$ , we follow (Avron et al., 2014) in exposing the algorithm fully with SVD. Note that both  $U$  and  $V$  correspond to a whitening step followed by a projection to a common space of dimension  $k$ . The total computational complexity of this algorithm assuming  $m_y < m_x$  is  $O(nm_x^2 + nm_y^2 + m_x m_y^2)$ . While the Bjork Golub SVD algorithm is intuitive and efficient, it is less popular in machine learning than the generalized eigenvalue implementation of CCA.

### 2.2. Bidirectional Retrieval with Asymmetrically Weighted CCA

As we have seen in Sections 2 and 2.1, the correlation operator  $T$  plays a central role in the least squares formulation as well in the CCA formulation. Turning back to the image to caption mapping and using the expressions of the whitened data points and the SVD of  $T$  given in Equation (2), we obtain:

$$f_{i \rightarrow c}(x^*) = T^\top u_x^* = P_y \Sigma P_x^\top \Sigma_x^{-1} V_x^\top x^* = P_y \Sigma U^\top x^*,$$

where  $U$  is the canonical weight for  $X$  and  $\Sigma$  the canonical correlation matrix. Note that  $f_{i \rightarrow c}$  is  $m_y$  dimensional. We can reduce its dimension by projecting it to the  $k$ -dimensional column space of  $P_y$ . It follows that:  $P_y^\top f_{i \rightarrow c}(x^*) = \Sigma U^\top x^* \in \mathbb{R}^k$ . On the other hand:  $P_y^\top u_{y^*} = P_y^\top \Sigma_y^{-1} V_y^\top y^* = V^\top y^* \in \mathbb{R}^k$ , where  $V$  is the canonical correlation weight for  $Y$ . For the image search problem, we can therefore perform the nearest neighbor search, in the  $k$ -dimensional reduced space defined by

$$(P_y^\top f_{i \rightarrow c}(x^*), P_y^\top u_{y^*}) = (\Sigma U^\top x^*, V^\top y^*).$$

It follows that the image search problem can be written using the canonical weights  $U, V$  and the canonical correlation  $\Sigma$ . Note that the correlation matrix is weighing the canonical weight in an asymmetric way. Cosine similarities are usually used for retrieval with CCA (Klein et al., 2015; Gong et al., 2014), hence we use cosine similarity between the two embeddings. The image search problem reduces to finding for a query caption  $y^*$ , the image  $x^*$  solving:

$$\arg \max_{x^* \in S_x^*} \frac{\langle \Sigma U^\top x^*, V^\top y^* \rangle}{\|\Sigma U^\top x^*\| \|V^\top y^*\|}. \quad (3)$$

Similarly using a cosine similarity the image annotation problem reduces to finding for a query image  $x^*$ , the cap-

tion  $y^*$  solving:

$$\arg \max_{y^* \in S_y^*} \frac{\langle U^\top x^*, \Sigma V^\top y^* \rangle}{\|U^\top x^*\| \|\Sigma V^\top y^*\|}. \quad (4)$$

Thus we see that  $\Sigma$  appears in an asymmetric way in the embedding of the points depending on the task. Hence we call our method asymmetrically weighted CCA (Table 2)

Task	Image Embedding	Caption Embedding
Search	$\Sigma U^\top x^*$	$V^\top y^*$
Annotation	$U^\top x^*$	$\Sigma V^\top y^*$

Table 2: Task dependent embeddings: Asymmetrically Weighted CCA.  $x^*$  is a test image,  $y^*$  is a test caption.  $(U, V)$  are the canonical weights of  $X$  and  $Y$ .  $\Sigma$  is the diagonal canonical correlations matrix. We use the cosine similarity for performing the search.

### 3. CCA Regularization

For now we have assumed that the covariances  $C_{XX}$  and  $C_{YY}$  are non-singular, and we presented an SVD version of the Bjork Golub Algorithm in this context. Regularizing CCA does not only allow for numerical stability in the non singular case, efficient model selection on a validation set allows for better generalization properties and avoids overfitting. Tikhonov regularization is the most common regularization used in CCA and consists in replacing the covariances by  $C_{XX} + \gamma_x I_{m_x}$  and  $C_{YY} + \gamma_y I_{m_y}$ , where  $\gamma_x, \gamma_y > 0$  are the regularization parameters subject to cross-validation. In this section we extend the SVD Bjork Golub Algorithm to the Tikhonov regularized CCA. Another contribution of this paper is in introducing the truncated SVD regularization to the covariances in the CCA problem. We emphasize that our truncation is applied to the SVD of the data matrices in the covariances of  $X$  and  $Y$ , not the SVD of the whitened correlation matrix  $T$  in the Bjork Golub Algorithm. Truncating the SVD of  $T$  and choosing an embedding dimension  $k < \min(m_x, m_y)$  is sometimes referred to as the truncated SVD CCA (Chaudhuri et al, 2009), hence our clarification. We show in the following how to specialize the Bjork Golub algorithm to handle T-SVD regularized CCA.

#### 3.1. Tikhonov and Truncated SVD Regularization

The Tikhonov regularized CCA problem for parameters  $\gamma_x, \gamma_y > 0$  can be written in this form:

$$\max_{U=I, V^\top(Y^\top Y + \gamma_y I_{m_y})V=I} Tr(U^\top X^\top Y V). \quad (5)$$

Let  $k_x \leq m_x$  and let  $X_{k_x}$  be the best  $k_x$ -rank approximation of  $X$  given by the truncated SVD:  $X_{k_x} = U_{k_x} \Sigma_{k_x} V_{k_x}^\top$ ,  $U_{k_x} \in \mathbb{R}^{n \times k_x}$ ,  $\Sigma_{k_x} \in \mathbb{R}^{k_x \times k_x}$ ,  $V_{k_x} \in$

$\mathbb{R}^{m_x \times k_x}$ . Similarly for  $Y$ , we define the best  $k_y$ -rank approximation ( $k_y \leq m_y$ ):  $Y_{k_y} = U_{k_y} \Sigma_{k_y} V_{k_y}^\top$ ,  $U_{k_y} \in \mathbb{R}^{n \times k_y}$ ,  $\Sigma_{k_y} \in \mathbb{R}^{k_y \times k_y}$ ,  $V_{k_y} \in \mathbb{R}^{m_y \times k_y}$ . We define the truncated SVD CCA as follows:

$$\max_{U=I, V^\top Y_{k_y}^\top Y_{k_y} V=I} Tr(U^\top X^\top Y V) \quad (6)$$

The following theorem shows how the Bjork-Golub procedure to compute the canonical weights extends to the regularized case, using singular value decompositions of the data matrices and a correlation operator.

**Theorem 1 (Regularized CCA).** *Let  $X = U_x \Sigma_x V_x^\top$ , and  $Y = U_y \Sigma_y V_y^\top$  be the singular value decomposition of  $X$  and  $Y$  ( $U_x \in \mathbb{R}^{n \times m_x}$ ,  $\Sigma_x \in \mathbb{R}^{m_x \times m_x}$ ,  $V_x \in \mathbb{R}^{m_x \times m_x}$ ). Let  $k = \min(m_x, m_y)$ .*

1) *Tikhonov Regularization. Define the Tikhonov regularized correlation operator*

$$T_{\gamma_x, \gamma_y} = (\Sigma_x^2 + \gamma_x I)^{-\frac{1}{2}} \Sigma_x (U_x^\top U_y) \Sigma_y (\Sigma_y^2 + \gamma_y I)^{-\frac{1}{2}},$$

*and let  $T_{\gamma_x, \gamma_y} = P_x \Sigma P_y^\top$  be its singular value decomposition ( $P_x \in \mathbb{R}^{m_x \times k}$ ,  $\Sigma \in \mathbb{R}^{k \times k}$ ,  $P_y \in \mathbb{R}^{m_y \times k}$ ). The canonical weights of the Tikhonov regularized CCA (5) are given by  $U = V_x (\Sigma_x^2 + \gamma_x I)^{-\frac{1}{2}} P_x$  and  $V = V_y (\Sigma_y^2 + \gamma_y I)^{-\frac{1}{2}} P_y$ . Canonical correlations are given by  $\Sigma$ .*

2) *Truncated SVD Regularization. Define the T-SVD regularized correlation operator  $T_{k_x, k_y} = U_{k_x}^\top U_{k_y}$ , and let  $T_{k_x, k_y} = P_x \Sigma P_y^\top$  be its singular value decomposition. The canonical weights of the T-SVD regularized CCA (6) are given by  $U = V_{k_x} \Sigma_{k_x}^{-1} P_x$  and  $V = V_{k_y} \Sigma_{k_y}^{-1} P_y$ . Canonical correlations are given by  $\Sigma$ .*

*Proof.* The proof is given in Appendix B.  $\square$

**Remark 1.** 1) While  $T_{k_x, k_y} \in \mathbb{R}^{k_x \times k_y}$  has a SVD computational cost of  $\min(O(k_x k_y^2), O(k_y k_x^2))$ ,  $T_{\gamma_x, \gamma_y} \in \mathbb{R}^{m_x \times m_y}$  and has a SVD computational cost of  $\min(O(m_x m_y^2), O(m_y m_x^2))$ . Hence T-SVD is more efficient computationally. 2) In T-SVD CCA the dimension of the embedding space is  $k = \min(k_x, k_y)$ . In the Tikhonov case, it is  $k = \min(m_x, m_y)$  (one can also compute a truncated SVD of  $T_{\gamma_x, \gamma_y}$  to reduce further the dimensionality of the embedding).

#### 3.2. Spectral Filtering for CCA

Note that in truncated SVD case we were computing the SVD of the regularized correlation operator:  $T_{k_x, k_y} = U_{k_x}^\top U_{k_y}$ . Define  $\sigma_{x,1}$  and  $\sigma_{x,m_x}$  minimum and maximum singular value of  $X$ . Let

$$T_f^{k_x, k_y} = f_{k_x}(\Sigma_x) U_x^\top U_y f_{k_y}(\Sigma_y), \quad (7)$$



where  $f_{k_x}$  is an element wise filter acting on the singular values of  $X$ , such that :

$$\begin{aligned} f_{k_x}(\sigma_{x,j}) &= 0 \text{ if } \sigma_{x,j} < \sigma_{x,k_x}, \\ \text{and } f_{k_x}(\sigma_{x,j}) &= 1 \text{ if } \sigma_{x,j} \geq \sigma_{x,k_x}. \end{aligned} \quad (8)$$

For a matrix  $A$ ,  $A(1 : k_x, 1 : k_y)$ , refers to the sub-matrix containing the first  $k_x$  rows and the first  $k_y$  columns of  $A$ . It is easy to see that:  $T_{k_x, k_y} = T_f^{k_x, k_y}(1 : k_x, 1 : k_y)$ . Turning now to the Tikhonov case, the regularized correlation operator is :

$$T_{\gamma_x, \gamma_y} = (\Sigma_x^2 + \gamma_x I)^{-\frac{1}{2}} \Sigma_x (U_x^\top U_y) \Sigma_y (\Sigma_y^2 + \gamma_y I)^{-\frac{1}{2}}$$

Let  $f_{\gamma_x}$  be also a spectral filter acting on the singular values of  $X$ , such that  $f_{\gamma_x, j} = \frac{\sigma_{x,j}}{\sqrt{\sigma_{x,j}^2 + \gamma_x}}$ ,  $j = 1 \dots m_x$ . Similarly we define  $f_{\gamma_y}$ . Define:

$$T_f^{\gamma_x, \gamma_y} = f_{\gamma_x}(\Sigma_x) U_x^\top U_y f_{\gamma_y}(\Sigma_y). \quad (9)$$

It is easy to see that  $T_f^{\gamma_x, \gamma_y} = T_{\gamma_x, \gamma_y}$ .

From Equations (7) and (9), we see that both regularization are proceeding by filtering of singular values. While T-SVD proceeds by a hard filtering, Tikhonov proceeds with a soft filtering. In order to see the correspondence between T-SVD and Tikhonov regularization it is important to consider the following choice of regularization parameters:

For a choice of  $(k_x, k_y)$  in T-SVD, consider  $\gamma_x = \sigma_{x,k_x}^2$  and  $\gamma_y = \sigma_{y,k_y}^2$  in Tikhonov Regularization.

With this particular choice the spectral filters for Tikhonov regularization become:

$$f_{\sigma_{k_x}^2}(\sigma_{x,j}) = \frac{\sigma_{x,j}}{\sqrt{\sigma_{x,j}^2 + \sigma_{x,k_x}^2}}. \quad (10)$$

To appreciate this particular choice of  $\gamma_x, \gamma_y$ , it is important to compare the spectral filter of T-SVD regularization for CCA given in Equation (8) and the spectral filter for Tikhonov Regularization for CCA given in Equation (10). Let  $\alpha > 0$ , consider:

- 1)  $g_{\text{hard}}(x) = 0$ , if  $0 \leq x < \alpha$ , and  $g_{\text{hard}}(x) = 1$  if  $x \geq \alpha$ .
- 2)  $g_{\text{soft}}(x) = \frac{x}{\sqrt{x^2 + \alpha^2}}$ .

It is easy to see that Equations (8) and (10), correspond to the element-wise application of  $g_{\text{hard}}$  and  $g_{\text{soft}}$  respectively for  $\alpha = \sigma_{k_x}$ . In Figure 1, we plot  $g_{\text{hard}}$  versus  $g_{\text{soft}}$ , for  $\alpha = 20$ . We see that both T-SVD and Tikhonov Regularization correspond to a spectral filtering of the singular values of  $X$  and  $Y$ . T-SVD is a hard pruning of directions corresponding to singular values less then the threshold defined by  $\sigma_{k_x}$ . For  $\alpha = \sigma_{k_x}$  Tikhonov corresponds to a soft pruning of those directions. In conclusion, using this spectral filtering approach the natural set of regularization parameters for Tikhonov regularization  $(\gamma_x, \gamma_y)$  is therefore the set of singular values squared of  $X$  and  $Y$  respectively. Hence in order to cross-validate CCA we propose the following three approaches:

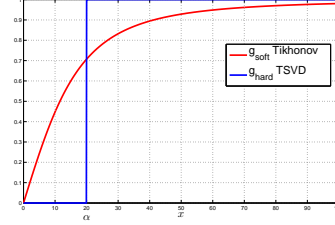


Figure 1: Spectral Filtering: We see in this that both T-SVD and Tikhonov Regularization correspond to a spectral filtering of the singular values of  $X$  and  $Y$ . T-SVD is a hard pruning of directions corresponding to singular values less than the threshold defined by  $\sigma_{k_x}$ . For  $\alpha = \sigma_{k_x}$  Tikhonov corresponds to a soft pruning of those directions.

1) **Tikhonov-CCA**: Perform a grid search on  $(\gamma_x, \gamma_y) \in \{\sigma_{1,x}^2, \dots, \sigma_{m_x,x}^2\} \times \{\sigma_{1,y}^2, \dots, \sigma_{m_y,y}^2\}$ , using the SVD of the regularized correlation operator  $T_{\gamma_x, \gamma_y}$ . Note that for Tikhonov cross-validation each SVD computation costs  $\min(O(m_x m_y^2), O(m_y m_x^2))$  (Algorithm 2 in Appendix C).

2) **T-SVD-CCA**: Perform a grid search on  $(k_x, k_y) \in [1, m_x] \times [1, m_y]$ , using the SVD of the regularized correlation operator  $T_{k_x, k_y}$ . This grid search (Algorithm 3 in Appendix C) is computationally efficient since each SVD computation costs  $\min(O(k_x k_y^2), O(k_y k_x^2))$ , hence more efficient than its Tikhonov counterpart. See Table 3 for CPU timing experiments on the MSCOCO Benchmark.

Table 3: We report the timing of computing regularization path of AW- CCA, with T-SVD and Tikhonov, on MSCOCO. The parameter grid for both cases is  $20 \times 20$ . Experiments were conducted on a single Intel Xeon CPU E5-2667, 3.30GHz, with 265 GB of RAM and 25.6 MB of cache. We see that T-SVD is faster to cross validate than Tikhonov : 2x speedup for VGG/w2v ( $m_x=4096$ ,  $m_y=300$ ), 6x speedup for VGG/skip thoughts ( $m_x=4096$ ,  $m_y=4800$ ).

AW- CCA	CPU time
T-SVD (VGG/w2v)	3763.654s
Tikhonov- CCA (VGG/w2v)	6981.697s
T-SVD (VGG/Skip)	24930.61s
Tikhonov (VGG/Skip)	163140.10s

3) **G-Tikhonov-CCA**: While there is no exact one to one correspondence between the soft pruning and the hard pruning in T-SVD and Tikhonov, the spectral filtering interpretation suggests that the optimal  $(k_x^*, k_y^*)$  from T-SVD cross-validation (computationally efficient), can be used as a proxy for the optimal Tikhonov cross validation

(computationally expensive), by simply setting  $(\gamma_x, \gamma_y) = (\sigma_{x,k_x}^2, \sigma_{y,k_y}^2)$  (See Table 4). This is summarized in Algorithm 4 in Appendix C. Algorithm 4, takes advantage of the fast computation of T-SVD regularization for CCA, and the spectral filtering interpretation in order to choose a good regularization parameter for Tikhonov Regularization.

AW-CCA	Image search			Image annotation		
	r@1	r@5	r@10	r@1	r@5	r@10
Tikh	<b>23.08</b>	<b>50.62</b>	<b>63.53</b>	<b>30.66</b>	<b>59.1</b>	<b>70.13</b>
T-SVD	20.61	46.91	59.9	27.0	55.5	67.80
G-Tikh	<b>22.40</b>	<b>49.94</b>	<b>62.62</b>	<b>30.56</b>	<b>58.43</b>	<b>69.96</b>

Table 4: Mean results (in %) of the test splits on the Flickr 30 K, in average w2vec/vgg setup. We see that G-Tikh and Tikh are on par.

#### 4. Hierarchical Kernel Sentence Embedding

**Bag of Words Distribution.** In this Section we define a new sentence embedding for aggregating local features in text description. Given a vocabulary  $\mathcal{A}$  represented by a vector space i.e a word embedding word2vec for instance (Mikolov et al., 2013), a sentence can be seen as a distribution  $\rho$  on  $\mathcal{A}$ .

**A Kernel between Distributions.** Given a distribution  $\rho$  defined on a vocabulary space  $\mathcal{A} \subset \mathbb{R}^d$ . Let  $k_{\gamma,d}$  be a shift invariant kernel such as the gaussian kernel with parameter  $\gamma$ , for  $a, b \in \mathcal{A}$ ,  $k_{\gamma,d}(a, b) = \exp(-\frac{\gamma}{2} \|a - b\|^2)$ . Let  $\mathcal{H}_k$  be the associated Reproducing Kernel Hilbert Space (RKHS), with norm  $\|\cdot\|_{\mathcal{H}_k}$ . The kernel mean embedding of  $\rho$  is defined (Smola et al, 2007; Sriperumbudur et al, 2010) as follows  $\mu(\rho) = \int_{\mathcal{A}} k_{\gamma,d}(a, \cdot) \rho(a) da$ .  $\mu \in \mathcal{H}_k$  and can be used to define a distance between two distributions  $\rho_1$  and  $\rho_2$  by means of  $\|\mu(\rho_1) - \mu(\rho_2)\|_{\mathcal{H}_k}^2$ . Given a finite sample  $\{a_1, \dots, a_n\}$  from  $\rho$  the empirical kernel mean embedding is given by:  $\mu_n(\rho) = \frac{1}{n} \sum_{i=1}^n k_{\gamma,d}(a_i, \cdot)$ . Let  $\eta > 0$ , we propose the following kernel between distributions  $\rho_1$  and  $\rho_2$  defined on  $\mathcal{A}$ , given a finite sample  $\{a_1 \dots a_{n_1}\}$  and  $\{b_1 \dots b_{n_2}\}$  from each distribution respectively<sup>2</sup>:

$$K(\rho_1, \rho_2) = \exp\left(-\frac{\eta}{2} \|\mu_{n_1}(\rho_1) - \mu_{n_2}(\rho_2)\|_{\mathcal{H}_k}^2\right).$$

$K$  can be seen also as a kernel between sets or bag of words kernel (Smola et al, 2007; Yuya et al, 2015; Yuya et al, 2014).  $K$  defines a universal kernel on distributions as shown in (Christmann al, 2010). Learning with  $K$  was analyzed in (Muandet et al, 2012).

##### Approximating $K$ with a Hierarchical Random Map

<sup>2</sup>It is easy to see that  $K(\rho_1, \rho_2) = \exp \frac{\eta}{2} (\sum_{i=1}^{n_1} \sum_{j=1}^{n_2} 2k_{\gamma,d}(a_i, b_j) - \sum_{i,j=1}^{n_1} k_{\gamma,d}(a_i, a_j) - \sum_{i,j=1}^{n_2} k_{\gamma,d}(b_i, b_j))$ .

Rather then using the kernel  $K$  and kernel CCA, that are computationally expensive, we make use of random Fourier features (Rahimi et al, 2007) in approximating  $K$  with an explicit feature map as in (Lopez-Paz et al., 2014). Let  $\Phi_{\gamma,d}(a) \in \mathbb{R}^m$ ,  $\Phi_{\gamma,d}(a) = (\cos(\langle w_1, a \rangle + b_1) \dots \cos(\langle w_m, a \rangle + b_m))$ ,  $w_j \sim \mathcal{N}(0, \gamma I_d)$ , and  $b \sim \text{Unif}[0, 2\pi]$ , we have  $\langle \Phi_{\gamma,d}(a), \Phi_{\gamma,d}(b) \rangle \approx k_{\gamma,d}(a, b)$ . Hence we define the randomized kernel mean map of  $\rho$ ,  $\hat{\mu}_n(\rho) = \frac{1}{n} \sum_{i=1}^n \Phi_{\gamma,d}(a_i)$ . For sufficiently large  $m$  we have

$$K(\rho_1, \rho_2) \approx k_{\eta,m} \left( \frac{1}{n_1} \sum_{i=1}^{n_1} \Phi_{\gamma,d}(a_i), \frac{1}{n_2} \sum_{i=1}^{n_2} \Phi_{\gamma,d}(b_i) \right).$$

$k_{\eta,m}$  is also a shift invariant kernel in  $m$  dimension, that can be in its turn approximated with a random feature map  $\Phi_{\eta,m} \in \mathbb{R}^{m'}$ . For sufficiently large  $m'$ , we have  $K(\rho_1, \rho_2) \approx$

$$\left\langle \Phi_{\eta,m} \left( \frac{1}{n_1} \sum_{i=1}^{n_1} \Phi_{\gamma,d}(a_i) \right), \Phi_{\eta,m} \left( \frac{1}{n_2} \sum_{i=1}^{n_2} \Phi_{\gamma,d}(b_i) \right) \right\rangle_{\mathbb{R}^{m'}}.$$

Hence  $K$  is approximated by a deep (2 layers) map, that is the composition of a non linear average pooling and a non linear feature map.

**Hierarchical Kernel Sentence Embedding (HKSE).** By embedding sentences consisting of word vectors (word2vec)  $a_1 \dots a_n$  using  $\Phi_{\eta,m}(\frac{1}{n} \sum_{i=1}^n \Phi_{\gamma,d}(a_i))$ , we compute implicitly the similarity  $K$  between the bag of words distributions. While  $k_{\gamma,d}$  or  $\Phi_{\gamma,d}$  act as a localizer at the word level,  $k_{\eta,m}$  or  $\Phi_{\eta,m}$  localize on the sentence level. Hence we note them respectively with  $k^w, \Phi^w$  (word level kernel), and  $k^s, \Phi^s$  (sentence level kernel). We note the embedding with  $\text{HKSE}(k^w, k^s)$ . The average word2vec representation proposed in (Klein et al., 2015) corresponds to  $\text{HKSE}(k^w, k^s)$ , where  $k^w$  and  $k^s$  are linear kernels.

**HKSE as a Gaussian Embedding of Sentences.** (Vilnis et al, 2015) introduced the embedding of words to Gaussian distributions (Word2Gauss), we show that similarly HKSE(lin,rbf) for  $k^w$  being linear and  $k^s$  being an rbf (radial basis function) kernel defines an embedding of sentences to Gaussian distributions (Sent2Gauss). Given  $\{a_1 \dots a_n\}$ , the vector embeddings of words in a sentence, we represent a sentence as a gaussian distribution  $\rho : \rho \sim \mathcal{N}(\mu, \sigma^2 I_d)$ ,  $\mu = \frac{1}{n} \sum_{i=1}^n a_i, \eta > 0$ . In order to compare two sentences we compare two distributions  $\rho_1$  and  $\rho_2$ , using product kernel between distributions (Jebara et al, 2003):  $K(\rho_1, \rho_2) = \int \mathcal{N}(x; \mu_1, \sigma^2 I_d) \mathcal{N}(x; \mu_2, \sigma^2 I_d) dx = \mathcal{N}(0; \mu_1 - \mu_2, 2\sigma^2 I_d) = (4\pi\sigma^2)^{-\frac{d}{2}} \exp\left(-\frac{1}{4\sigma^2} \|\mu_1 - \mu_2\|^2\right)$ , which corresponds to the kernel computed by  $\text{HKSE}(\text{lin}, \text{rbf})$ .  $\text{HKSE}(\text{rbf}, \text{rbf})$  corresponds also to a universal kernel between distributions (Christmann al, 2010).

## 5. Relation to Previous work

We focus in this section on some recent works that use bidirectional maps in the retrieval tasks and their relation to this paper. (Klein et al., 2015), and (Gong et al., 2014) used CCA to build a joint representation using the cosine similarity. In both works a symmetric weighting of the CCA canonical weights was used, i.e for an image caption pair  $(x^*, y^*)$ , a joint embedding of the form  $(\Sigma^\alpha U^\top x^*, \Sigma^\alpha V^\top y^*)$ , was used, where  $\alpha = 0$ , in (Klein et al., 2015) case and  $\alpha > 0$  in (Gong et al., 2014) case. The symmetric weighting in (Gong et al., 2014) was a heuristic found to improve performance and is not theoretically motivated as in the asymmetric weighting of this paper. Indeed our experimental results show that asymmetric weighting gives higher performance. A discussion of the optimality of the asymmetric weighting and an empirical study is given in Appendix D. Skip thought vectors introduced in (Kiros et al., 2015) for representing sentences were used for bidirectional search in conjunction with VGG features (Simonyan & Zisserman, 2015) on the image side, with linear embeddings learned with a discriminative triplets loss, instead of a CCA loss. A similar discriminative loss was used with linear features in (Lebret et al., 2015).

## 6. Numerical Results

**Datasets.** We performed image annotation and search tasks on the MSCOCO benchmark (Lin et al., 2014), and Flickr 8K and 30 K benchmarks (Hodosh et al., 2013; Young et al, 2014) using our task dependent asymmetrically weighted CCA, as described in Table 2. Retrieval was performed using cosine similarities given in Equations (3) and (4). For MSCOCO the training set contains 113,287 images, along with 5 captions each. Similarly to (Klein et al., 2015), we used the splits from (Karpathy & Li, 2015), and performed cross-validation on a validation set of 5K images, and tested our models on a test set of 5K images, as well as five 1K splits of the 5K test images as in (Klein et al., 2015). We report for both tasks the recall rate at one result, five results, or ten first results ( $r@1,5,10$ ), as well as the median rank of the first ground truth retrieval.

We follow the experimental protocol of

[github.com/ryankiros/skip-thoughts/blob/master/eval\\_rank.py](https://github.com/ryankiros/skip-thoughts/blob/master/eval_rank.py)

For 5K test images, we have 25K query captions, the image search retrieval scoring is based on the ground truth image of each caption. For image annotation, we have 5K query test images, to annotate among 25K captions, the scoring returns the caption that has the highest cosine within each 5 captions. A similar scoring is done for 1K tests.

The Flickr 8K training set contains 6K images along with

5 captions each, the validation and test set contain 1K images each. We use the splits as specified in (Hodosh et al., 2013). The Flickr 30K (Young et al, 2014) training set contains 25381 images along with 5 captions each, the validation and test sets contain 3K images each. We follow (Karpathy & Li, 2015) and use 3 random splits of 1K images for test and validation and report average performance over three runs.

**AW-G-Tikh Validation.** Cross validation was performed using the Asymmetrically Weighted TSVD-Guided-Tikhonov (AW-G-Tikh-CCA) given in Algorithm 4 (Appendix C) to select the model corresponding to the maximum  $r@1$  on the validation set (regularization paths are given in Appendix E).

**Image Features.** For feature extraction we follow (Klein et al., 2015), and use on the image side the VGG CNN representation (Simonyan & Zisserman, 2015), where each image was rescaled to have smallest side 384 pixels, and then cropped in 10 ways into 224 by 224 pixel images: the four corners, the center, and their x-axis mirror image. The mean intensity of each crop is subtracted in each color channel, and then encoded by VGG19 (the final FC -4096 layer). The average of the resulting 10 feature vectors corresponding to each crop is used as the image representation.

**Text Features.** On the text side, we use two sentence embeddings 1) HKSE( $k^w, k^s$ ), introduced in this paper, using word2vec available on [code.google.com/p/word2vec/](https://code.google.com/p/word2vec/) (throughout our experiments,  $\gamma$  was set to be the inverse of the squared median pairwise distances of words in the vocabulary,  $\eta$  was fixed to 0.01,  $m = 1000$  and  $m' = 2000$ ). Note that HKSE(lin,lin), corresponds to the mean word2vec baseline as in (Klein et al., 2015). 2) Skip thought vectors introduced in (Kiros et al., 2015), which encodes sentences to vectors using an LSTM. Image and text features were centered before learning CCA.

**Discussion of the Results.** We make the following comments: 1) We see in Tables 5 and 6, that AW-G-Tikh-CCA with the same Mean Vec features outperforms consistently the CCA baseline in (Klein et al., 2015), this mainly due to the asymmetric weighting and the query generation approach as discussed in Section 2, as well as the efficient model selection introduced in this paper. 2) HKSE(rbf,rbf) consistently outperforms the mean vector baseline as well as skip thoughts vectors and can be used therefore as a simple yet strong off the shelf sentence embedding. 3) HKSE(rbf,rbf) outperforms the fisher vector representation (Klein et al., 2015), on the MSCOCO Benchmark. Note that, Fisher vectors are learned on the dataset, and HKSE is fully unsupervised.

	Image search				Image annotation			
	r@1	r@5	r@10	med r	r@1	r@5	r@10	med r
1K test images:								
CCA+ Mean Vec (Klein et al., 2015)	24.2	56.4	72.4	4.0	33.2	61.8	75.1	3.0
AW-G-Tikh-CCA + HKSE(lin,lin) (Mean Vec)	28.14	61.33	76.64	3.0	37.16	68.78	81.36	2.0
AW-G-Tikh-CCA+ HKSE (lin,rbf)	30.99	65.43	79.49	3.0	39.94	71.82	84.1	2.0
AW-G-Tikh-CCA+ HKSE (rbf,lin)	29.54	63.79	78.70	3.0	41.10	70.90	82.46	2.0
AW-G-Tikh-CCA + HKSE(rbf,rbf)	<b>32.70</b>	<b>67.51</b>	<b>81.14</b>	<b>3.0</b>	43.64	74.94	85.68	2.0
AW-G-Tikh-CCA + HKSE(rbf,rbf) (Ann Pack)	NA	NA	NA	NA	<b>55.12</b>	<b>86.36</b>	<b>94.24</b>	<b>1.0</b>
Skip thoughts +AW-G-Tikhonov-CCA	29.14	63.74	77.53	3.0	39.24	70.44	82.68	2.0
Skip thoughts +Triplets loss (Kiros et al., 2015)	25.9	60	74.6	NA	33.8	67.7	82.1	NA
BRNN (Karpathy & Li, 2015)	20.9	52.8	69.2	4.0	29.4	62.0	75.9	2.5
Fisher GMM+HGLMM+CCA (Klein et al., 2015)	25.6	60.4	76.8	4.0	38.9	68.4	80.1	2.0
5K test images:								
CCA + Mean Vec (Klein et al., 2015)	10.3	27.2	38.4	18.0	12.8	32.1	44.6	14.0
AW-G-Tikh-CCA + HKSE(lin,lin) (Mean Vec)	12.91	32.17	44.24	14.0	17.9	40.30	53.12	9.0
AW-G-Tikh-CCA+ HKSE (lin,rbf)	14.24	35.39	48.34	11.0	19.06	43.02	57.02	8.0
AW-G-Tikh-CCA+ HKSE (rbf,lin)	13.46	33.74	46.08	13.0	19.98	44.50	56.90	7.0
AW-G-Tikh-CCA+ HKSE (rbf,rbf)	<b>15.44</b>	<b>37.44</b>	<b>50.69</b>	<b>10.0</b>	22.14	47.76	60.68	6.0
AW-G-Tikh-CCA + HKSE(rbf,rbf) (Ann Pack)	NA	NA	NA	NA	<b>32.46</b>	<b>62.20</b>	<b>75.00</b>	<b>3.0</b>
Skip thoughts + AW-G-Tikhonov-CCA	12.83	33.73	47.04	12.0	18.5	42.24	55.34	8.0
Skip thoughts +Triplets loss (Kiros et al., 2015)	NA	NA	NA	NA	NA	NA	NA	NA
BRNN (Karpathy & Li, 2015)	8.9	24.9	36.3	19.5	11.8	32.5	45.4	12.2
Fisher GMM+HGLMM + CCA (Klein et al., 2015)	11.2	29.2	41.0	16.0	17.7	40.1	51.9	10.0

Table 5: Mean results of the test splits on the MSCOCO benchmark (in %). Numbers in bold refer to our method.

	Image search				Image annotation			
	r@1	r@5	r@10	med r	r@1	r@5	r@10	med r
Flickr 30 K								
CCA + Mean Vec (Klein et al., 2015)	20.5	46.3	59.3	6.8	24.8	52.5	64.3	5.0
AW-G-Tikh-CCA+ HKSE (lin,lin) (Mean Vec)	22.40	49.94	62.62	6.0	30.56	58.43	69.96	4.0
AW-G-Tikh-CCA+ HKSE (rbf,rbf)	<b>25.80</b>	<b>54.72</b>	<b>66.52</b>	<b>4.0</b>	32.5	61.03	73.23	3.0
AW-G-Tikh-CCA + HKSE(rbf,rbf) (Ann Pack)	NA	NA	NA	NA	<b>43.1</b>	<b>74.4</b>	<b>84.3</b>	<b>2.0</b>
AW-G-Tikhonov-CCA + Skip thoughts	22.18	48.9	60.92	6.0	28.23	56.50	68.36	4.0
BRNN (Karpathy & Li, 2015)	15.2	37.7	50.5	9.2	22.2	48.2	61.4	4.8
Fisher GMM+HGLMM+CCA (Klein et al., 2015)	25.0	52.7	66.0	5.0	35.0	62.0	73.80	3.0
Flickr 8K								
CCA + Mean Vec (Klein et al., 2015)	19.1	45.3	60.4	7.0	22.6	48.8	61.2	6.0
AW-G-Tikh-CCA+ HKSE (lin,lin) (Mean Vec)	18.62	44.82	58.88	7.0	23.10	50.8	63.00	5.0
AW-G-Tikh-CCA+ HKSE (rbf,rbf)	19.6	45.66	58.00	7.0	25.5	53.4	67.60	5.0
AW-G-Tikh-CCA + HKSE(rbf,rbf) (Ann Pack)	NA	NA	NA	NA	<b>36.1</b>	<b>68.7</b>	<b>79.4</b>	<b>2.0</b>
Skip thoughts + AW-G-Tikhonov-CCA	17.52	43.76	57.92	7.0	21.70	50.20	63.70	5.0
BRNN (Karpathy & Li, 2015)	11.8	32.1	44.7	12.4	16.5	40.6	54.2	7.6
Fisher GMM+HGLMM + CCA (Klein et al., 2015)	<b>21.2</b>	<b>50.0</b>	<b>64.8</b>	<b>5.0</b>	31.0	59.3	73.7	4.0

Table 6: Mean results of the test splits on the Flickr 30 K and 8K benchmarks (in %). Numbers in bold refer to our method.

**Retrieving A Caption Set.** We consider the problem of assigning to each image the set of five ground truth captions. In order to return a single caption we select within the returned set the caption with largest cosine. Note that for MSCOCO and Flickr datasets  $S = \{(x_i, \{y_{i,j}, j = 1 \dots 5\}), i = 1 \dots N\}$  the correlation matrix :  $C_{XY} = \sum_{i=1}^N x_i \sum_{j=1}^5 y_{i,j}^\top$ . Hence the CCA objective is naturally correlating the image  $x_i$  with the unnormalized average caption  $\sum_{j=1}^5 y_{i,j}^\top$ . Hence the natural representation of the set of five captions by their average. As more words are available HKSE(rbf,rbf) gets a better estimate of the under-

lying distribution and achieves state of the art performance on annotation (Ann Pack in Tables 5 and 6 ).

## 7. Conclusion

In this paper we showed that the query generation approach in information retrieval (Lafferty et al, 2002), can be applied to bidirectional retrieval, where we map the search space to the query space via an asymmetric weighting of Regularized CCA. Asymmetric weighting improves the performance of the bidirectional retrieval tasks. We



also presented a computationally efficient cross validation for regularized CCA, that allows for a better model selection and hence contributes also in improving the retrieval performance. Finally we presented the Hierarchical Kernel Sentence Embedding that is of independent interest, and that generalizes the mean word2vec as a mean for aggregation of word embeddings and outperforms off the shelf sentence embeddings in bidirectional retrieval. We solved in this paper CCA in its batch formulation, for handling larger scale datasets we can use the randomized SVD of (Halko et al., 2011), the subsampled CCA of (Avron et al., 2014), or SGD CCA of (Ma et al., 2015; Wang et al., 2015).

## References

- Avron, Haim, Boutsidis, Christos, Toledo, Sivan, and Zouzias, Anastasios. Efficient dimensionality reduction for canonical correlation analysis. *SIAM J. Scientific Computing*, 2014.
- Fang, Hao, Gupta, Saurabh, Iandola, Forrest N., Srivastava, Rupesh K., Deng, Li, Dollr, Piotr, Gao, Jianfeng, He, Xiaodong, Mitchell, Margaret, Platt, John C., Zitnick, C. Lawrence, and Zweig, Geoffrey. From captions to visual concepts and back. In *CVPR*, 2015.
- Farhadi, Ali, Hejrati, Seyyed Mohammad Mohsen, Sadeghi, Mohammad Amin, Young, Peter, Rashtchian, Cyrus, Hockenmaier, Julia, and Forsyth, David A. Every picture tells a story: Generating sentences from images. In *ECCV (4)*, volume 6314, 2010.
- Golub, Gene H., Björck, He. Numerical methods for computing angles between linear subspaces. *Math. Comp.*, 1973.
- Gong, Yunchao, Wang, Liwei, Hodosh, Micah, Hockenmaier, Julia, and Lazebnik, Svetlana. Improving image-sentence embeddings using large weakly annotated photo collections. In *ECCV*, 2014.
- Halko, N., Martinsson, P. G., and Tropp, J. A. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.*, 2011.
- Hansen, Per C. The truncated svd as a method for regularization. Technical report, 1986.
- Hotelling, H. Relations between two sets of variates. *Biometrika*, 1936.
- Karpathy, Andrej and Li, Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015.
- Karpathy, Andrej, Joulin, Armand, and Li, Fei-Fei. Deep fragment embeddings for bidirectional image sentence mapping. In *NIPS*, 2014.
- Kiros, Ryan, Salakhutdinov, Ruslan, and Zemel, Richard S. Unifying visual-semantic embeddings with multimodal neural language models. *CoRR*, 2014.
- Kiros, Ryan, Zhu, Yukun, Salakhutdinov, Ruslan, Zemel, Richard S., Torralba, Antonio, Urtasun, Raquel, and Fidler, Sanja. Skip-thought vectors. *NIPS*, 2015.
- Klein, Benjamin, Lev, Guy, Sadeh, Gil, and Wolf, Lior. Associating neural word embeddings with deep image representations using fisher vectors. In *CVPR*, 2015.

- Kulkarni, Girish, Premraj, Visruth, Dhar, Sagnik, Li, Siming, Choi, Yejin, Berg, Alexander C., and Berg, Tamara L. Baby talk: Understanding and generating simple image descriptions. In *CVPR*, 2011.
- Lebret, Rmi, Pinheiro, Pedro O., and Collobert, Ronan. Phrase-based image captioning. In *ICML*, 2015.
- Lin, Tsung-Yi, Maire, Michael, Belongie, Serge, Hays, James, Perona, Pietro, Ramanan, Deva, Dollr, Piotr, and Zitnick, C. Lawrence. Microsoft MSCOCO: Common objects in context. In *ECCV*, 2014.
- Lopez-Paz, D., Sra, S., Smola, A., Ghahramani, Z., and Schölkopf, B. Randomized nonlinear component analysis. In *Proceedings of the 31st International Conference on Machine Learning, W&CP 32 (1)*, pp. 1359–1367. JMLR, 2014.
- Ma, Zhuang, Lu, Yichao, and Foster, Dean P. Finding linear structure in large datasets with scalable canonical correlation analysis. In *ICML*, 2015.
- Mao, Junhua, Xu, Wei, Yang, Yi, Wang, Jiang, and Yuille, Alan L. Explain images with multimodal recurrent neural networks. *CoRR*, abs/1410.1090, 2014.
- Mikolov, Tomas, Chen, Kai, Corrado, Greg, and Dean, Jeffrey. Efficient estimation of word representations in vector space. *arXiv preprint*, 2013.
- Simonyan, Karen and Zisserman, Andrew. Very deep convolutional networks for large-scale image recognition. *ICLR*, 2015.
- Socher, Richard, Lin, Cliff Chiung-Yu, Ng, Andrew Y., and Manning, Christopher D. Parsing natural scenes and natural language with recursive neural networks. In *ICML*, 2011.
- Socher, Richard, Ganjoo, Milind, Manning, Christopher D., and Ng, Andrew. Zero-shot learning through cross-modal transfer. In *Advances in Neural Information Processing Systems 26*, pp. 935–943. 2013.
- Socher, Richard, Karpathy, Andrej, Le, Quoc V., Manning, Christopher D., and Ng, Andrew Y. Grounded compositional semantics for finding and describing images with sentences. *TACL*, 2014.
- Vinod, H. D. Canonical ridge and econometrics of joint production. *Journal of Econometrics*, 1976.
- Vinyals, Oriol, Toshev, Alexander, Bengio, Samy, and Erhan, Dumitru. Show and tell: A neural image caption generator. In *CVPR*, 2015.
- Wang, Weiran, Arora, Raman, Srebro, Nati, and Livescu, Karen. Stochastic optimization for deep cca via nonlinear orthogonal iterations. In *Allerton Conference*, 2015.
- M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: data, models and evaluation metrics. In *Journal of Artificial Intelligence Research*, 2013.
- P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. In *TACL*, 2014.
- Kamalika Chaudhuri, Sham M. Kakade, Karen Livescu, and Karthik Sridharan. Multi-View Clustering via Canonical Correlation Analysis In *ICML*, 2009.
- Friedman J, Hastie T and Tibshirani R. Regularization Paths for Generalized Linear Models via Coordinate Descent In *Journal of statistical software*, 2010.
- Alex Smola, Arthur Gretton, Le Song, and Bernhard Scholkopf. A Hilbert Space Embedding for Distributions In *Algorithmic Learning Theory*, 2007.
- Bharath K. Sriperumbudur, Arthur Gretton, Kenji Fukumizu, Bernhard Scholkopf, and Gert R.G. Lanckriet. Hilbert Space Embeddings and Metrics on Probability Measures. In *The Journal of Machine Learning Research*, 2010.
- Ali Rahimi and Ben Recht Random features. for large-scale kernel machines. In *NIPS*, 2007
- Yoshikawa Yuya , Iwata Tomoharu , Sawada Hiroshi and Yamada Takesh. Cross-Domain Matching for Bag-of-Words Data via Kernel Embeddings of Latent Distributions. In *NIPS*, 2015
- Yuya Yoshikawa, Tomoharu Iwata, and Hiroshi Sawada. Cross-Domain Matching for Bag-of-Words Data via Kernel Embeddings of Latent Distributions. In *NIPS*, 2014
- Andreas Christmann and Ingo Steinwart Universal Kernels on Non-Standard Input Spaces. In *NIPS*, 2010.
- Muandet, Krikamol and Kenji Fukumizu and Francesco Dinuzzo and Schölkopf, Bernhard. Learning from Distributions via Support Measure Machines. In *NIPS*, 2012.
- Luke Vilnis and Andrew McCallum Word Representations Via Gaussian Embedding. In *ICLR*, 2015.
- Tony Jebara and Risi Kondor Bhattacharyya and Expected Likelihood Kernels In *COLT*, 2003.

John Lafferty and Chengxiang Zhai Probabilistic Relevance Models Based on Document and Query Generation. In *Language Modeling and Information Retrieval*, 2002.

Francis R. Bach and Michael I. Jordan A probabilistic interpretation of canonical correlation analysis. In *Tech report*, 2005.

## Supplementary Material

### A. Appendix: Query Generation Versus Search Generation

(Lafferty et al, 2002) propose to use a binary random variable  $r$  that denotes relevance between a query and a search item,  $r = 1$  if there is a match and 0 otherwise. In order to rank search items (Lafferty et al, 2002) propose the use of the log-odds ratio:

$$\log \frac{\mathbb{P}(r = 1|q, s)}{\mathbb{P}(r = 0|q, s)}$$

Using the search generation approach this ratio is equivalent to (Lafferty et al, 2002):

$$\log \frac{\mathbb{P}(s|q, r = 1)}{\mathbb{P}(s|q, r = 0)},$$

hence this approach needs both positives and negatives pairs of search and query items (triplet losses are instances of this approach). On the other hand using the query generation approach (under mild assumptions) this ratio is equivalent to (Lafferty et al, 2002):

$$\log \mathbb{P}(q|s, r = 1)$$

The query generation approach models relevance in an implicit way and does not need negative samples.

### B. Appendix: Proofs

*Proof of Lemma 1.* We give the proof here as it will be useful in the the development of the full regularization path of CCA with truncated SVD regularization of the covariances  $C_{XX}$  and  $C_{YY}$ . Let  $P_x = \Sigma_x V_x^\top U \in \mathbb{R}^{m_x \times k}$  equivalently  $U = V_x \Sigma_x^{-1} P_x$  and  $P_y = \Sigma_y V_y^\top V \in \mathbb{R}^{m_y \times k}$  equivalently  $V = V_y \Sigma_y^{-1} P_y$ . Hence we obtain by this change of variable:

$$U^\top X^\top Y V = P_x^\top \Sigma_x^{-1} V_x^\top V_x \Sigma_x U_x^\top U_y \Sigma_y V_y^\top V_y \Sigma_y^{-1} P_y = P_x^\top (U_x^\top U_y) P_y.$$

Similarly:  $U^\top X^\top X U = P_x^\top P_x$   $V^\top Y^\top Y V = P_y^\top P_y$ . Hence replacing  $U, V$  with  $P_x, P_y$  we have:

$$\max_{P_x^\top P_x = I, P_y^\top P_y = I} \text{Tr}(P_x^\top (U_x^\top U_y) P_y),$$

This is solved by an SVD of  $T = U_x^\top U_y$ .  $[P_x, \Sigma, P_y] = \text{SVD}(T)$ , ( $P_x \in \mathbb{R}^{m_x \times k}, \Sigma \in \mathbb{R}^{k \times k}, P_y \in \mathbb{R}^{m_y \times k}$ ). where  $k = \min(m_x, m_y)$ , and finally we have  $U = V_x \Sigma_x^{-1} P_x$ ,  $V = V_y \Sigma_y^{-1} P_y$ .  $\square$

*Proof of Theorem 1.*

1) Tikhonov:

$$\max_{U^\top (X^\top X + \gamma_x I_{m_x}) U = I, V^\top (Y^\top Y + \gamma_y I_{m_y}) V = I} \text{Tr}(U^\top X^\top Y V). \quad (11)$$

$$[U_x, \Sigma_x, V_x] = \text{SVD}(X) \quad U_x \in \mathbb{R}^{n \times m_x}, \Sigma_x \in \mathbb{R}^{m_x \times m_x}, V_x \in \mathbb{R}^{m_x \times m_x} \quad X = U_x \Sigma_x V_x^\top.$$

$$[U_y, \Sigma_y, V_y] = \text{SVD}(Y) \quad U_y \in \mathbb{R}^{n \times m_y}, \Sigma_y \in \mathbb{R}^{m_y \times m_y}, V_y \in \mathbb{R}^{m_y \times m_y} \quad Y = U_y \Sigma_y V_y^\top.$$

$$X^\top X + \gamma_x I = V_x (\Sigma_x^2 + \gamma_x I) V_x^\top.$$

$$Y^\top Y + \gamma_y I = V_y (\Sigma_y^2 + \gamma_y I) V_y^\top.$$

Let  $P_x = \sqrt{\Sigma_x^2 + \gamma_x I} V_x^\top U \in \mathbb{R}^{m_x \times k}$  equivalently  $U = V_x (\Sigma_x^2 + \gamma_x I)^{-\frac{1}{2}} P_x$ . Let  $P_y = \sqrt{\Sigma_y^2 + \gamma_y I} V_y^\top V \in \mathbb{R}^{m_y \times k}$  equivalently  $V = V_y (\Sigma_y^2 + \gamma_y I)^{-\frac{1}{2}} P_y$ . Hence we obtain by this change of variable for the objective in (11):

$$\begin{aligned} U^\top X^\top Y V &= P_x^\top (\Sigma_x^2 + \gamma_x I)^{-\frac{1}{2}} V_x^\top V_x \Sigma_x U_x^\top U_y \Sigma_y V_y^\top V_y (\Sigma_y^2 + \gamma_y I)^{-\frac{1}{2}} P_y \\ &= P_x^\top (\Sigma_x^2 + \gamma_x I)^{-\frac{1}{2}} \Sigma_x (U_x^\top U_y) \Sigma_y (\Sigma_y^2 + \gamma_y I)^{-\frac{1}{2}} P_y. \end{aligned}$$



Let

$$T_{\gamma_x, \gamma_y} = (\Sigma_x^2 + \gamma_x I)^{-\frac{1}{2}} \Sigma_x (U_x^\top U_y) \Sigma_y (\Sigma_y^2 + \gamma_y I)^{-\frac{1}{2}},$$

hence we have:

$$U^\top X^\top Y V = P_x^\top T_{\gamma_x, \gamma_y} P_y.$$

On the other hand, plugging this change of variable in the constraints of (11) we obtain:

$$U^\top (X^\top X + \gamma_x I) U = U^\top V_x (\Sigma_x^2 + \gamma_x I) V_x^\top U = P_x^\top P_x = I$$

$$V^\top (Y^\top Y + \gamma_y I) V = V^\top V_y (\Sigma_y^2 + \gamma_y I) V_y^\top V = P_y^\top P_y = I$$

Therefore using this change of variable, problem (11) becomes:

$$\max_{P_x^\top P_x = I, P_y^\top P_y = I} Tr(P_x^\top T_{\gamma_x, \gamma_y} P_y), \quad (12)$$

this is the variational formulation of the SVD of  $T_{\gamma_x, \gamma_y}$ . Hence we obtain that:

$$[P_x, \Sigma, P_y] = SVD(T_{\gamma_x, \gamma_y}),$$

$$U = V_x (\Sigma_x^2 + \gamma_x I)^{-\frac{1}{2}} P_x,$$

$$V = V_y (\Sigma_y^2 + \gamma_y I)^{-\frac{1}{2}} P_y.$$

2) T-SVD:

$$\max_{U^\top X_{k_x}^\top X_{k_x} U = I, V^\top Y_{k_y}^\top Y_{k_y} V = I} Tr(U^\top X^\top Y V) \quad (13)$$

Let

$$P_x = \Sigma_{k_x} V_{k_x}^\top U \in \mathbb{R}^{k_x \times k}, \text{ equivalently } U = V_{k_x} \Sigma_{k_x}^{-1} P_x \in \mathbb{R}^{m_x \times k}, k = \min(k_x, k_y)$$

$$P_y = \Sigma_{k_y} V_{k_y}^\top V \in \mathbb{R}^{k_y \times k}, \text{ equivalently } V = V_{k_y} \Sigma_{k_y}^{-1} P_y \in \mathbb{R}^{m_y \times k}, k = \min(k_x, k_y)$$

Hence we obtain by this change of variable, in the objective of (13):

$$U^\top X^\top Y V = P_x^\top \Sigma_{k_x}^{-1} V_{k_x}^\top V_x \Sigma_x U_x^\top U_y \Sigma_y V_y^\top V_{k_y} \Sigma_{k_y}^{-1} P_y,$$

Now we turn to:

$$\begin{aligned} \Sigma_{k_x}^{-1} (V_{k_x}^\top V_x) \Sigma_x U_x^\top &= \Sigma_{k_x}^{-1} [I_{k_x \times k_x} \ 0_{k_x \times (m_x - k_x)}] \Sigma_x U_x^\top \\ &= \Sigma_{k_x}^{-1} [\Sigma_{k_x} \ 0_{k_x \times (m_x - k_x)}] U_x^\top \\ &= [I_{k_x \times k_x} \ 0_{k_x \times (m_x - k_x)}] U_x^\top \\ &= U_{k_x}^\top \end{aligned}$$

Hence we keep the first  $k_x$  columns of  $U_x$ , that is  $U_{k_x}$ . The same argument hold for  $U_{k_y}$ . It follows that using truncated SVD, we have:

$$U^\top X^\top Y V = P_x^\top U_{k_x}^\top U_{k_y} P_y.$$

Let

$$T_{k_x, k_y} = U_{k_x}^\top U_{k_y},$$

then using this change of variable, the objective in (13) becomes:

$$U^\top X^\top Y V = P_x^\top T_{k_x, k_y} P_y.$$

Now turning to the constraints of (13), using this change of variable we obtain:

$$U^\top X_{k_x}^\top X_{k_x} U = U^\top V_{k_x} \Sigma_{k_x}^2 V_{k_x}^\top U = P_x^\top P_x = I,$$

$$V^\top Y_{k_y}^\top Y_{k_y} V = V^\top V_{k_y} \Sigma_{k_y}^2 V_{k_y}^\top V = P_y^\top P_y = I.$$

Therefore using this change of variable, problem (13) becomes:

$$\max_{P_x^\top P_x = I, P_y^\top P_y = I} \text{Tr}(P_x^\top T_{k_x, k_y} P_y), \quad (14)$$

this is the variational formulation of the SVD of  $T_{k_x, k_y}$ . Hence truncated SVD-CCA can be solved finding:

$$[P_{k_x}, \Sigma^{k_x, k_y}, P_{k_y}] = \text{SVD}(T_{k_x, k_y}).$$

Turning now to  $U_{k_x}^\top U_{k_y}$  this can be computed efficiently by precomputing  $T = U_x^\top U_y \in \mathbb{R}^{m_x \times m_y}$  and then extracting the submatrix consisting of  $k_x$  rows and  $k_y$  columns. we return therefore  $U = V_{k_x} \Sigma_{k_x}^{-1} P_x$ ,  $V = V_{k_y} \Sigma_{k_y}^{-1} P_y$ .  $\square$

## C. Algorithms

---

### Algorithm 1 Bjork Golub

---

- 1:  $[U_x, \Sigma_x, V_x] = \text{SVD}(X)$ .
  - 2:  $[U_y, \Sigma_y, V_y] = \text{SVD}(Y)$ .
  - 3:  $T = U_x^\top U_y \in \mathbb{R}^{m_x \times m_y}$
  - 4:  $[P_x, \Sigma, P_y] = \text{SVD}(T)$
  - 5:  $U = V_x \Sigma_x^{-1} P_x$
  - 6:  $V = V_y \Sigma_y^{-1} P_y$
  - 7: return  $U, V$
- 

---

### Algorithm 2 Tikhonov Regularized CCA (X,Y)

---

- 1:  $[U_x, \Sigma_x, V_x] = \text{SVD}(X)$ .
  - 2:  $[U_y, \Sigma_y, V_y] = \text{SVD}(Y)$ .
  - 3:  $T_0 = \Sigma_x (U_x^\top U_y) \Sigma_y \in \mathbb{R}^{m_x \times m_y}$
  - 4: **for**  $\gamma_x \in \{\sigma_{x,1}^2, \dots, \sigma_{x,m_x}^2\}$  **do**
  - 5:   {The set of singular values squared or a subsampled grid}
  - 6:   **for**  $\gamma_y \in \{\sigma_{y,1}^2, \dots, \sigma_{y,m_y}^2\}$  **do**
  - 7:      $T = (\Sigma_x^2 + \gamma_x I)^{-\frac{1}{2}} T_0 (\Sigma_y^2 + \gamma_y I)^{-\frac{1}{2}}$
  - 8:      $[P_x, \Sigma^{\gamma_x, \gamma_y}, P_y] = \text{SVD}(T)$
  - 9:      $U = V_x (\Sigma_x^2 + \gamma_x I)^{-\frac{1}{2}} P_x$
  - 10:     $V = V_y (\Sigma_y^2 + \gamma_y I)^{-\frac{1}{2}} P_y$
  - 11:    Compute performance using  $U, V, \Sigma^{\gamma_x, \gamma_y}$  on a validation Set.
  - 12:    (Bidirectional Retrieval is done using a sorted list of the scores in Eqs (3), and (4).)
  - 13:   **end for**
  - 14: **end for**
  - 15: return  $U, V, \Sigma^{\gamma_x, \gamma_y}, \gamma_x, \gamma_y$  with best validation performance for each task.
-

---

**Algorithm 3** Truncated SVD CCA (X,Y)
 

---

```

1:  $[U_x, \Sigma_x, V_x] = SVD(X)$ .
2:  $[U_y, \Sigma_y, V_y] = SVD(Y)$ .
3:  $T = U_x^\top U_y \in \mathbb{R}^{m_x \times m_y}$ 
4:  $W^x = V_x \Sigma_x^{-1} \in \mathbb{R}^{m_x \times m_x}$ 
5:  $W^y = V_y \Sigma_y^{-1} \in \mathbb{R}^{m_y \times m_y}$ 
6: for  $k_x \in [m_x]$  do
7:    $\{[m_x] = \{1 \dots m_x\} \text{ or a subsampled grid.}\}$ 
8:   for  $k_y \in [m_y]$  do
9:      $[P_x, \Sigma^{k_x, k_y}, P_y] = SVD(T_{1:k_x, 1:k_y})$ 
10:     $\{T_{1:k_x, 1:k_y} : \text{extracts the first } k_x \text{ rows, and the first } k_y \text{ columns of } T.\}$ 
11:     $U = W^x_{:, 1:k_x} P_x$ 
12:     $V = W^y_{1:k_y, :} P_y$ 
13:    Compute performance using  $U, V, \Sigma^{k_x, k_y}$  on a validation Set.
14:    (Bidirectional Retrieval is done using a sorted list of the scores in Eqs (3), and (4).)
15:   end for
16: end for
17: return  $U, V, \Sigma^{k_x, k_y}, k_x, k_y$  with best validation performance for each task.
```

---



---

**Algorithm 4** Guided Tikhonov Validation by T-SVD(X,Y)
 

---

```

1:  $(k_x^*, k_y^*) = \text{Truncated SVD CCA}(X, Y)$ 
2:  $(\gamma_x, \gamma_y) = (\sigma_{x, k_x^*}^2, \sigma_{y, k_y^*}^2)$ 
3:  $[P_x, \Sigma^{\gamma_x, \gamma_y}, P_y] = SVD(T_{\gamma_x, \gamma_y})$ 
4:  $U = V_x (\Sigma_x^2 + \gamma_x I)^{-\frac{1}{2}} P_x$ ,
5:  $V = V_y (\Sigma_y^2 + \gamma_y I)^{-\frac{1}{2}} P_y$ 
6: return  $U, V, \Sigma^{\gamma_x, \gamma_y}$ .
```

---

## D. Optimality of the task dependent asymmetric weighting

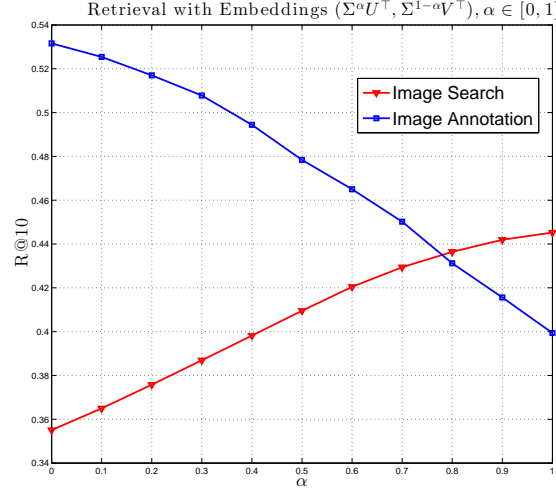


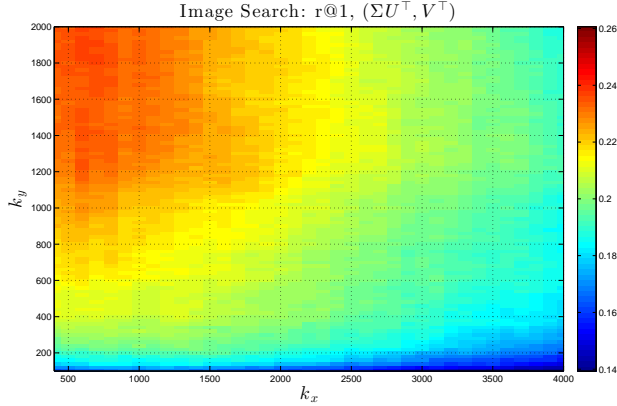
Figure 2: Optimality of the asymmetric weighting: This plot shows R@10 versus  $\alpha$  on the validation set, for image annotation and image search performed using the image embedding  $\Sigma^\alpha U^\top$  and caption embedding  $\Sigma^{1-\alpha} V^\top$ , where  $\alpha \in [0, 1]$ . The cosine similarity was used between embeddings. In this experiment we used VGG image features and average word2vec sentence embedding. For each  $\alpha$  we perform a thorough cross validation using truncated SVD CCA, and report the best R@10 for image annotation and image search. We see that  $\alpha = 0$ , i.e the embedding  $(U^\top, \Sigma V^\top)$  is optimal for image annotation as predicted by the least squares formulation for image annotation in this paper.  $\alpha = 1$ , i.e the embedding  $(\Sigma U^\top, V^\top)$ , is optimal for image search as predicted by the least squares formulation.

	Image search			Image annotation		
	r@1	r@5	r@10	r@1	r@5	r@10
Symmetric Weighting Heuristic (Gong et al., 2014)						
$(\Sigma^\alpha U^\top, \Sigma^\alpha V^\top), \alpha = 0$ (CCA)	9.36	26.13	37.30	13.96	34.30	47.10
$(\Sigma^\alpha U^\top, \Sigma^\alpha V^\top), \alpha = 1$	11.21	28.95	40.24	13.72	34.00	45.86
$(\Sigma^\alpha U^\top, \Sigma^\alpha V^\top), \alpha = 2$	10.04	26.54	37.46	11.16	29.02	39.56
$(\Sigma^\alpha U^\top, \Sigma^\alpha V^\top), \alpha = 3$	8.30	23.53	33.78	9.36	24.74	35.56
$(\Sigma^\alpha U^\top, \Sigma^\alpha V^\top), \alpha = 4$	7.09	20.79	30.81	7.88	21.58	32.22
$(\Sigma^\alpha U^\top, \Sigma^\alpha V^\top), \alpha = 5$	6.16	18.63	27.79	6.76	19.76	29.46
$(\Sigma^\alpha U^\top, \Sigma^\alpha V^\top), \alpha = 6$	5.55	16.76	25.31	6.08	17.80	27.12
Task Dependent Asymmetric Weighting:						
Image Search: $(\Sigma U^\top, V^\top)$ , Image Annotation: $(U^\top, \Sigma V^\top)$	12.92	32.38	44.52	17.84	40.42	53.16

Table 7: Comparison to Symmetric Weighting Heuristic of (Gong et al., 2014): In the VGG/Word2vec setup, we perform a thorough cross validation using T-SVD CCA for embeddings of the form  $(\Sigma^\alpha U^\top, \Sigma^\alpha V^\top)$ ,  $\alpha \geq 0$ , used jointly with the cosine similarity for retrieval tasks r@1,5,and 10. Results are reported here on the validation set.  $\alpha = 0$  corresponds to using the CCA weights as in (Klein et al., 2015). We see that this symmetric weighting boosts the search performance for a particular  $\alpha = 1$  as reported in (Gong et al., 2014). The asymmetric weighting we propose outperforms the symmetric weighting heuristic as shown in this table, and boosts performance of both tasks.

## E. AWT-SVD CCA Regularization Paths





(a) T-SVD Cross Validation.

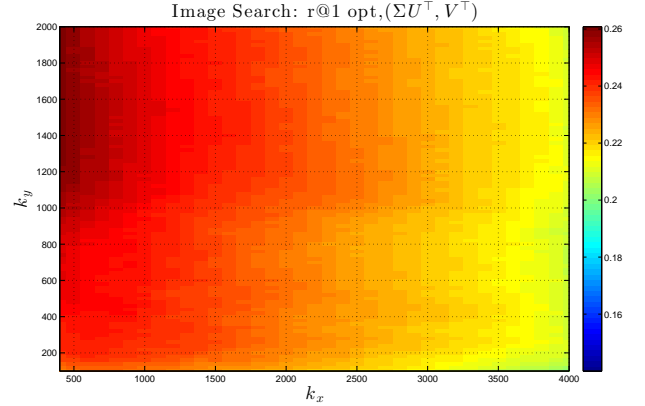
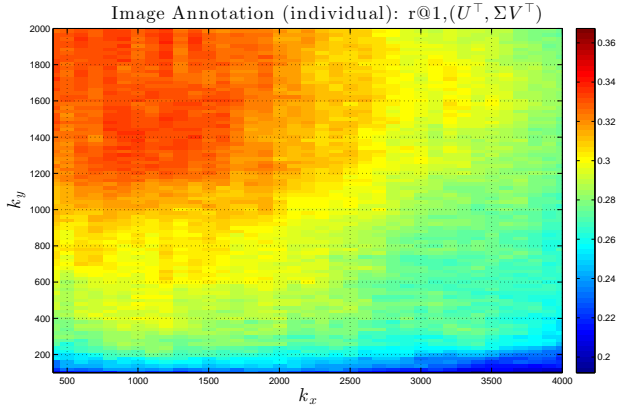

 (b) Tikhonov Cross Validation  
 $(\gamma_x = \sigma_{k_x}^2, \gamma_y = \sigma_{k_y}^2).$ 

Figure 3: Regularization Path for T-SVD CCA , and Tikhonov CCA on bidirectional retrieval on Flickr30K with VGG features (4096 dimensions) for the image and HKSE(rb,rbf) (2000 dimensions). Cross validation was performed on the validation set on grid going from 400 to 4000 with step size of 100 for  $k_x$ , and from 200 to 2000 with a step size of 20 for  $k_y$  . We report r@1 of the retrieved query over the validation set (Higher is better, in red). We see that T-SVD and Tikhonov select the same region of interest, justifying the T-SVD guided Tikhonov approach.



(a) T-SVD Cross Validation.

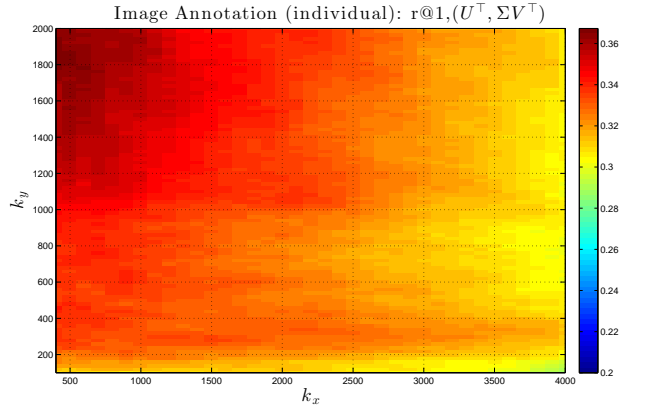

 (b) Tikhonov Cross Validation  
 $(\gamma_x = \sigma_{k_x}^2, \gamma_y = \sigma_{k_y}^2).$ 

Figure 4: Regularization Path for T-SVD CCA , and Tikhonov CCA on bidirectional retrieval on Flickr30K with VGG features (4096 dimensions) for the image and HKSE(rb,rbf) (2000 dimensions). Cross validation was performed on the validation set on grid going from 400 to 4000 with step size of 100 for  $k_x$ , and from 200 to 2000 with a step size of 20 for  $k_y$  . We report r@1 of the retrieved query over the validation set (Higher is better, in red). We see that T-SVD and Tikhonov select the same region of interest, justifying the T-SVD guided Tikhonov approach.

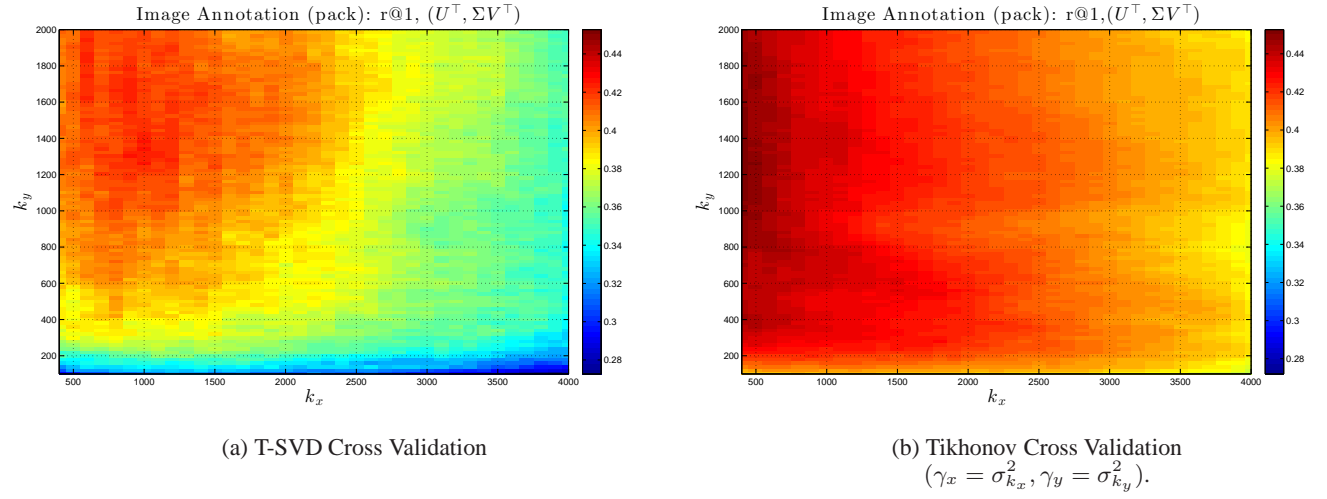


Figure 5: Regularization Path for T-SVD CCA , and Tikhonov CCA on bidirectional retrieval on Flickr30K with VGG features (4096 dimensions) for the image and HKSE(rb,rbf) (2000 dimensions). Cross validation was performed on the validation set on grid going from 400 to 4000 with step size of 100 for  $k_x$ , and from 200 to 2000 with a step size of 20 for  $k_y$  . We report r@1 of the retrieved query over the validation set (Higher is better, in red). We see that T-SVD and Tikhonov select the same region of interest, justifying the T-SVD guided Tikhonov approach.